**Learner’s Academy**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Submitted By:

Hrushikesh Kalamkar

([hrushilp18@gmail.com](mailto:hrushilp18@gmail.com))

GitHub-

|  |  |  |
| --- | --- | --- |
| **Sr.No** | **Contents** | **Page No.** |
| 1 | **Objective and background of the statement** | **3** |
| **2** | **Flowchart** | **4** |
| **3** | **Sprint** | **5** |
| **4** | **Technologies/Concepts used and requirements** | **6** |

**Project Objective:**

As a Full Stack Developer, design and develop a backend administrative portal for the Learner’s Academy.

Use the GitHub repository to manage the project artifacts.

**Background of the problem statement:**

Learner’s Academy is a school that has an online management system.

The system keeps track of its classes, subjects, students, and teachers. It has a back-office application with a single administrator login.

Flowchart



Sprint Planning

Sprint 1:

1. Create a database
2. Create Table in the database.
3. Connect the database to the backend using JDBC.
4. Create model classes to retrieve data from the database.
5. Create Login page.

Sprint 2:

1. Create JSP files for all pages of the project.
2. Create servlets to fetch requests and send responses to the JSP files.
3. Format the contents by creating a CSS file.
4. Debug
5. Test the project

**Technologies/Concepts used and requirements**

**Concepts used:**

1. Object-Oriented: used to create and model objects for users and their credentials.
2. Databases: used to store and retrieve data.
3. Data Sources: used to define a set of properties required to identify and access the database.
4. Collections: used some collections such Arraylist to store collection of data.
5. Exception Handling: used to catch problems that arises in the code especially in I/O blocks.
6. Cookies: to store log-in data on the client browser. Searching and Sorting.

**Technologies Used:**

1. Eclipse
2. Java
3. MYSQL
4. Git
5. GitHub
6. Scrum
7. Search and Sort techniques
8. Specification Document

Requirements:

1. The source code should be pushed to your GitHub repository. You need to document the steps and write the algorithms in it.
2. The submission of your GitHub repository link is mandatory. In order to track your task, you need to share the link of the repository. You can add a section in your document.
3. Document the process step-by-step starting from sprint planning to the product release.
4. The application should not close, exit, or throw an exception if the user specifies an invalid input.
5. You need to submit the final specification document which will include:
6. Project and developer details
7. Sprints planned and the tasks achieved in them
8. Algorithms and flowcharts of the application
9. Core concepts used in the project
10. Links to the GitHub repository to verify the project completion